NPS-CS-11-009

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

Leveraging the Cloud to Support Communications in the Tactical Environment

by

James Bret Michael, George Dinolt, Doron Drusinsky,
Loren Peitso, Thomas Otani, Man-Tak Shing

15 December 2011

Approved for public release; distribution is unlimited

Prepared for: Office of the Under Secretary of Defense for Intelligence,
Intelligence, Surveillance, and Reconnaissance
5000 Defense Pentagon, Washington, DC 20301-5000

THIS PAGE INTENTIONALLY LEFT BLANK

NAVAL POSTGRADUATE SCHOOL
Monterey, California 93943-5000

Daniel T. Oliver
President

Leonard A. Ferrari
Executive Vice President and
Provost

The report entitled "Leveraging the Cloud to Support Communications in the Tactical Environment" was prepared for and funded by the Office of the Under Secretary of Defense for Intelligence, Intelligence, Surveillance, and Reconnaissance.

Further distribution of all or part of this report is authorized.

This report was prepared by:

James Bret Michael
Professor, Computer Science and
Electrical Engineering

George Dinolt
Professor of Practice
Computer Science

Doron Drusinsky
Associate Professor
Computer Science

Loren Peitso
Senior Lecturer
Computer Science

Thomas Otani
Associate professor
Computer Science

Man-Tak Shing
Associate Professor
Computer Science

Reviewed by:

Peter J. Denning, Chairman
Computer Science Department

Released by:

Karl A. van Bibber
Vice President and
Dean of Research

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
15 December 2011	Technical Report	16 May 2011 – 30 Sept. 2011

4. TITLE AND SUBTITLE
Leveraging the Cloud to Support Communications in the Tactical Environment

5a. CONTRACT NUMBER
DSAM10523

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
James Bret Michael, George Dinolt, Doron Drusinsky, Loren Peitso, Thomas Otani, and Man-Tak Shing

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
1411 Cunningham Road
Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER
NPS-CS-11-009

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of the Under Secretary of Defense for Intelligence – Intelligence, Surveillance, and Reconnaissance

5000 Defense Pentagon, Washington, DC 20301-5000

10. SPONSOR/MONITOR'S ACRONYM(S)
OUSD(I)/ISR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES
The views expressed in this report are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

14. ABSTRACT

Tactical computing includes all computations necessary to provide shared situational awareness among geographically dispersed forces in a digitally connected battlespace. Cloud computing, which builds on grid computing, service-oriented computing, and virtualization technologies, is an enabler for building the globally distributed system-of-systems necessary for accomplishing the ultimate goal of net-centric warfare. Moreover, secured and robust tactical communications are essential for the optimum use of cloud techniques to distribute data and processing over the tactical environment. This report presents two high-level use cases to understand how cloud computing can play a role in supporting spectrum management—an important component of tactical communications—and presents key challenges and possible approaches to address the challenges in making the cloud-based system timely, robust, trustworthy, and interoperable, as well as a means to validate and verify such systems.

15. SUBJECT TERMS
Cloud computing, Spectrum management, Tactical radio, Cellular radio

16. SECURITY CLASSIFICATION OF: UNCLASSIFIED			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			James Bret Michael
Unclassified	Unclassified	Unclassified	UU	22	**19b. TELEPHONE NUMBER** *(include area code)* 571-858-3253

THIS PAGE INTENTIONALLY LEFT BLANK

1. INTRODUCTION

Tactical computing involves the processing of sensor data to produce timely information for warfighters to use in making effective decisions, and the accurate and timely dissemination of orders (including weapons control data) to various units and systems during an engagement. With the advent of net-centric warfare, which first appeared publicly in the article by Cebrowski and Garstka in 1998 [1] and was further refined in the articles by Alberts et al. [2, 3], tactical computing now includes all computations necessary to provide shared situational awareness among geographically dispersed forces in a digitally connected battlespace. To accomplish the ultimate goal of net-centric warfare, we need a globally distributed system-of-systems that allows the "edge-entities" who are conducting military missions to "smart-pull" information from ubiquitous sources at anytime and from anywhere. Cloud computing, which builds on grid computing, service-oriented computing, and virtualization technologies, is an enabler for building such globally distributed system-of-systems. While cloud computing holds promise to help improve workflows and efficiency in the military strategic, operational and tactical level defense processes, the tactical environment's special limitations and constraints pose some especially challenging engineering problems to be solved.

In FY2010, we identified some of the technical enablers for applying cloud computing in tactical systems. We developed high-level generic use cases to understand how cloud computing can play a role in supporting changes in the workflows employed by warfighters to attain information superiority [4]. From those use cases, we identified the following topical areas as being of high priority for further investigation: (i) interoperability among hybrid clouds, (ii) timeliness of data, computation, and communication, (iii) system safety, (iv) system security, (v) continuity of operations, (vi) dynamic reconfiguration, and (vii) phased migration. We also identified trustworthy and robust tactical communications as an essential enabler for the optimum use of cloud techniques to distribute data and processing over the tactical environment.

This report summarizes the research conducted in FY11. We focus on the potential benefits of cloud computing on tactical communications, which involves the transmission of both voice and digital data via multiple waveforms (e.g., cellular and RF) over a wide spectrum of radio frequencies (i.e., from RF to SHF).

2. SPECTRUM MANAGEMENT

In [5], Dixon identifies opportunities for leveraging commercial cellular technology in military communications systems by integrating a variety of wireless technologies without compromising security and reliability. The US Department of Defense (DoD) is experimenting with the hybrid mobile devices that accommodate both tactical radio and cellular waveforms. In this report we present our vision of using the cloud to ubiquitously supply services to the tactical edge.

Spectrum management involves optimizing the use of the different waveforms and frequencies to maximize mission effectiveness while minimizing interference. The Spectrum-Management system is a distributed system designed to coordinate and manage

the use of frequencies and waveforms to maximize communication between mobile devices (e.g., tactical radios, smartphones, tablet computers) and access points of the tactical cloud. It consists of a set of regional spectrum-management services. Each regional spectrum-management service is a Software-as-a-Service (SaaS) and is accessible from mobile devices and the regional tactical communication providers via the tactical cloud access points in the corresponding communication region. The communication regions may overlap, in which case these regional spectrum-management services collaborate with each other to manage the overlapped areas.

The following subsections contain two use cases highlighting the intended use of the spectrum-management services.

2.1 Use Case 1 – Spectrum Selection

Actors: User, Regional Spectrum-Management Service

Description: This use case describes the process for the applications on a user's mobile device to acquire frequency and waveform allocations from the regional spectrum-management service.

Main Scenario:

(1) User activates a tactical cloud-based application function that, in its operation, will require spectrum use.

(2) The tactical cloud-based application informs the Spectrum Manager (SM) application on the mobile device (MD) of the necessary communications requirements (e.g., in terms of bandwidth, security level, and media types).

(3) The SM application on the MD authenticates itself to the Regional Spectrum-Management cloud service and the Regional Spectrum-Management Service authenticates itself to the SM application on the device and the two establish a secure (encrypted) channel that provides data-integrity checks, all done over the default frequency and waveform. All further communications that use the default frequency and waveform take place over a secure channel.[1]

(4) The SM application on the mobile device negotiates with the Regional Spectrum-Management Service via the default frequency and waveform on behalf of the tactical cloud-based application for allocation of spectrum assets.

(5) The Regional Spectrum-Management Service replies with the recommended frequency and waveform, together with alternate frequencies and waveforms.

(6) The SM application informs the tactical cloud-based application on the MD of the available frequency and waveform allocations.

(7) The tactical cloud-based application uses the information provided by the SM to inform the user of the MD of the application quality-of-service (QoS) the tactical

[1] Note that if this is not done, then an adversary could potentially become a "man-in-the-middle" and/or manipulate the assignment of resources to its advantage.

cloud-based application will be able to provide based upon the actual frequency and waveform availability.

(8) Advanced users can be presented the opportunity to manually update the spectrum requests (e.g., in terms of bandwidth, security level, and media types).[2] This will result in reentering Step 3 to perform additional spectrum-negotiation.

(9) The user accepts the tactical cloud-based application QoS and executes the intended application functionality. The MD uses the negotiated frequency and waveform allocations and opens a communication session.

Exceptions:

(3a) If default frequency not available, the SM notifies the user of the MD and initiates a search for available frequencies and waveforms.

2.2 Use Case 2 – Spectrum Management

Actors: Regional Spectrum-Management Service, Regional Tactical Communication Provider, User

Description: This use case describes the process for the Regional Spectrum-Management Service to coordinate and manage the use of frequencies and waveforms to maximize communication between the mobile devices and the access points of the tactical cloud.

Main Scenario:

(1) The Regional Tactical Communication Providers (RTCP) inform the Regional-Spectrum-Management Service about the health status and current/scheduled usage of available frequencies and waveforms.

(2) The Regional Spectrum-Management Service updates its regional tactical communication common operating picture (RTC COP).

(3) A SM application on the user's MD provides the Regional Spectrum-Management Service with communication requirements.

(4) The Regional Spectrum-Management Service cross references the request with its RTC COP, returns the frequency and waveform allocation to the MD, and updates its RTC COP.

(5) The user's MD initiates the communication session via the use of the tactical cloud-based application.

(6) The User, the tactical cloud-based application, and the SM of the MD monitor the communication session to ensure that the negotiated QoS is maintained. In the event of service degradation (e.g., due to traffic congestion or poor reception), the

[2] Advanced users here means users that have an appropriate level of training to perform such actions, such a Navy Information Systems Technician or Army Signal Corps Signal Support Systems Specialist or Telecommunications Systems Engineer.

SM and the Regional Spectrum-Management Service negotiate new allocations of frequencies and waveforms and the Regional Spectrum-Management Service updates the RTC COP.

(7) The RTCP notifies the Regional Spectrum-Management Service about problems with any of the managed frequencies and waveforms. The Regional Spectrum-Management Service updates the RTC COP and then sends updated frequency and waveform allocations to the applicable participants in communication session.

(8) The User terminates a communication session via the tactical cloud-based application, completing the task that required communication. Upon task completion, the tactical cloud-based application notifies the MD's SM application, which in turn notifies the Spectrum-Management Service that the MD is relinquishing its frequency and waveform allocation. The Spectrum-Management Service updates its RTC COP.

3. VERIFICATION & VALIDATION CONSIDERATIONS

The Spectrum-Management system is a heterogeneous distributed system with the following agents:

(1) Tactical communication provider (plurality)[3]

(2) SM (singleton per-user device)

(3) Regional Spectrum-Management Service SaaS (plurality)

(4) User device (plurality)

3.1 The Verification of Distributed Systems - An Overview

Distributed systems are notoriously hard to verify. While classical, academic techniques such as Theorem Proving (TP) and Model Checking (MC) can be used to verify rather simple correctness properties (assertions) with respect to small parts of the system, or to a highly abstracted model of the system, Runtime Verification (RV) remains the preferred method for verifying a large, distributed, (soft) real-time system. See [6] for further details.

RV is a hybrid of formal specification (e.g., using UML statechart assertions) and conventional testing, where the actual behavior of the System Under Test (SUT) is compared automatically with its expected behavior as specified by the assertions. This is done by executing an implementation version of the formal specification assertions in tandem with the SUT (i.e., on-line RV), or by executing the assertions against a recorded trace of the SUT's execution (i.e., off-line, perhaps remote, RV).

Verification of distributed systems is notoriously difficult because of the following aspects:

[3] There will be places on the Earth where communications providers overlap. Depending on who "owns" various satellite channels this may be almost everywhere.

(1) *Complexity*. A distributed systems' state space is extremely large (being a Cartesian product of the state space of its constituent systems). Consequently, it is difficult to cover the state space in a trustworthy way during testing.

(2) *Timing*. Constituent systems within a distributed system do not, for the most part, share a common clock. Consequently, temporal assertions that assert about timing of more than a single sub-system are difficult to verify.

(3) *Contractual issues*. By contract it is meant the "promise" a component or sub-systems makes with respect to its interface, namely, the manner in which it will respond given certain input sequences received via its interface, and the constraints it is expecting those inputs to conform to. Distributed systems often fail to formally specify contracts, rendering verification all but impossible.

(4) *Accurate representation.* Fielded distributed systems do not behave like their lab, or development version counter parts. This is because of the dynamic aspect of the system, and the fact that the composite behavior of a fielded distributed system tends to be different than its lab version.

3.2 Suggested Verification Strategy for SM

We suggested a contract-based and prototype-based RV approach for the verification of the Spectrum-Management system. The proposed contract based RV requires four sets of requirements:

(1) *Interface requirements*. This is a well-defined set of contracts specifying the inter-component communication format and behavior. For example, consider the interface requirement for a Regional Spectrum-Management Service component, per Step 4 of UC1: *data packages received from SM via default frequency and waveform may not exceed 1K bytes in length and will be received at most one packet per 100ms interval*.

(2) *Sub*-system *requirements*. Each is a set of requirements for the particular sub-system.

(3) *Plurality requirements*. A (rather limited) set of requirements discussing aspects of the system that deal with the existence of a plurality of subsystems, such as "*If a user send communication requirements to Regional Spectrum-Management Service A, it may not, within a 15 second interval, send communication requirements to Regional Spectrum-Management Service B*".

Sub-system requirements are verified in a two-step process:

(1) Sub-system verification. Here, each subsystem is verified on a standalone basis.

(2) System level verification, which consists of the verification of interface and plurality requirements. This step requires an executable model or platform, on which the model of the distributed system is verified using RV. The proposed

approach is to rapidly develop a model of the distributed system, abstracted so it models the interface contracts (using sub-system requirements, that is, assuming the sub-systems had already been verified in Step 1).

We have rapidly developed a similar a prototype in the past, as a proof-of-concept demonstration, using the Eclipse/OSGi framework (http://eclipse.org/osgi/).

4. SECURITY CONSIDERATIONS

We can view the system under consideration as a resource management system. There are a number of application layer security concerns (e.g., application requirements for "quality of security service") that will not be directly represented by the security discussion below.

The correct operation of the system depends on the availability of a frequency/waveform for communication for use when requested. In addition, the fact that a set of waveforms and frequencies may be used or that a particular set of frequencies is in use shall not be available to unauthorized parties. As a result one could have the following security objectives:

(1) Only authorized equipment will be allowed to connect to the system, make requests for resources and utilize the communications resources.

(2) Requests for resource allocation shall be handled based on a priority scheme to be established by the users of the system.[4]

(3) Information about allocation of a resource shall not be available to unauthorized entities.

(4) The data and programs used to make decisions about allocation shall only be modifiable by authorized entities.

(5) The computer programs that are used to make the resource allocation decisions shall be validated that they operate correctly.

(6) The resource allocation system shall provide audit of all transactions.

4.1 Only Authorized Equipment

This is intended to prevent unauthorized devices from accessing the system. One way of implementing this is to provide a name for each device and a means of mutual authentication between the device and the resource-allocation entity. This would (hopefully) prevent misuse of the spectrum resources by our forces and the enemy from masquerading as a legitimate endpoint. There will also need to be a means of preventing

[4] We recognized that a set of advanced users might, with appropriate authorization, manually flood the system with requests for allocation of the same or even several resources. If advanced users are given this authorization, then the "fair" allocation of resources becomes a "personnel management" issue.

the enemy from operating captured devices. This could be implemented by rekeying all access points.[5]

4.2 Resource Allocation Policy

The intent here is that the owners of the system will provide policy that provides for the allocation of resources based on priority scheme. The system should have programs that implement this policy. Such policy might, for example, provide some users access to the resource, even if it is fully utilized, by removing users with lower priority. It may reallocate resources by removing capabilities from one user and reallocating them to another based on, for example use of the resource. The job of the resource management system is to enforce the users resource policy.

4.3 Privacy of Resource Data

The intent of this objective is to ensure that the data used to allocate resources and which resources are allocated to which entities are only "visible" to the resource manager and to authorized parties. For example, unauthorized parties should not be able to see which resource is in use.

4.4 Data and Program Protection

The goal here is to ensure the integrity of both the data being used to make decisions and the programs that use that data for actually making and enforcing the decisions. For example, we need to be able to ensure that only the correct, authorized programs are being run, that an unauthorized party cannot make changes to the system.

4.5 Programs are Correct

The goal of this objective is to ensure, to the level required by the resource owners, that the resource management system will operate correctly. This can be accomplished by sound engineering practices, where appropriate, the application of mathematical methods to show correctness, and various forms of testing.

4.6 Transactions are Audited

We see there are two uses for the audit data. One is to ensure that the system is behaving as specified by policy. The second use is to provide resources for investigation of potential misuse of the system. Enough audit data must be collected to support both of these uses. Policies need to be established on the preservation of audit data and its integrity.

[5] Some of the compromised-radio issues will be addressed by protocols from existing hardware programs.

5. TIMELINES OF DATA, COMPUTATION, AND COMMUNICATION CONSIDERATIONS

The discussion of timelines within distributed systems is concerned with continuity and preservation, that is, the extent to which data, computation and communication are consistent and continuous over time.

In the client-server world of the Web, a computation is in fact manifested as a single entity over a fairly long timeline using the notion of a *session*. Web programmers using techniques such as Java Servlets actually have access to a session object - an object that encapsulated a computation between a specific client (client being a browser activation instance, or even a browser tab instance) and the server. Using the session object the programmer can persist the state of the computation as the client traverses pages; the state of user authentication being one obvious example of such persisted state.

While a Web session captures the client-server conversation along the timeline segment that starts with the launch of the client and its termination, some Web users use settings in which the timeline is longer, namely persistence exists beyond the session. For example, a persistent cookie can preserve information about the user's website-traversal history for a period as long as a year.

In a cloud setting, timeline-persistence information such as the session information can be either stored by the client or stored in the cloud.

Cloud-side timeline management is expected to be difficult to use in conjunction with load-balancing implementations, because the session overhead data makes it difficult to move the session from one web server to another. The load-balancing problem can be solved by using shared storage or by applying forced peering between each client and a single server in the cloud, techniques that run contrary to the goal of load-balancing.

Cloud-side session management has an advantage being capable to aborting a session when it discovers suspicious security-related information about the client, such as the client's GPS data (in the case of a mobile client) indicating that the client is not where it is supposed to be. Also, cloud-side timeline management provides the infrastructure that enables fusion of data provided by a large group of users.

Client-side session management removes load-balancing limitations discussed above.

Client-side sessions use cookies and cryptographic techniques to maintain state without storing as much data on the cloud. However, when refreshing the client's data (e.g., new Web page), the following communication must take place: (i) the cloud sends the current state data to the client (e.g., in the form of a cookie), (ii) the client saves the cookie in memory or on disk, (iii) the client sends the cookie back to the server upon its next request, and the server uses the data to re-incarnate its memory of the state of the application for that specific client, thereby generating an appropriate response (Step (i), again).

Clearly, data stored on the client poses a security risk because the mobile device might be lost by its user or captured by enemy forces. Two security requirements needed for addressing this risk are:

(1) Confidentiality: Nothing apart from the server should be able to interpret session data.

(2) Data integrity: Nothing apart from the server should manipulate session data (accidentally or maliciously).

6. INTEROPERABILITY CONSIDERATIONS

Interoperability has been recognized as a major issue in net-centric operations, and it will remain a major issue in spectrum management. We need to address both data interoperability across organization boundaries of the local tactical cloud, and other supporting private clouds, such as the Intelligence and Maritime domain clouds, and data/application/virtualization interoperability across cloud providers' actual equipment installed. Ongoing DoD efforts, like the US Navy's Consolidated Afloat Networks and Enterprise Services (CANES) program and the Army's Network Integration Evaluation (NIE) process, are designed to streamline and update DoD tactical networks to improve interoperability across the fleet and services.

A widely recognized model for system-of-systems interoperability is the Levels of Information System Interoperability (LISI) Maturity Model published by the Department of Defense (DoD) C4ISR Architecture Working Group [7]. LISI classifies the degree of sophistication with respect to exchanging and sharing information and services among systems in terms of PAID, an acronym for four closely interrelated attributes: Procedures, Applications, Infrastructure, and Data, where:

(1) The procedures (P) attribute reflects the degree of interoperability resulting from operational policies and processes, functional program development guidance, as well as compliance of technical and system architecture standards (e.g., hardware, system software, communications, data, and application standards).

(2) The application (A) attribute reflects the ability of the software applications to work on different systems and platforms as they progress through the interoperability maturity levels, ranging from stand-alone applications at the low end to applications that are designed for cross-discipline or cross-organizational boundaries at the high end.

(3) The infrastructure (I) attribute reflects the degree and form of connectivity between the systems and applications (e.g., point-to-point phone connection versus wide-area network across great variety of systems and communication protocols), and the way in which the systems interact with each other (e.g., application specific interface versus platform independent Web services).

(4) The data (D) attribute reflects the flexibility of the data format and the richness of the information being exchanged across systems and domains

(ranging from files containing a single data type to integrated information space that supports all forms of data representation, presentation, and exploitation).

The LISI model focuses on system-to-system information exchanges but falls short in providing a basis for assessing the maturity of cloud-to-cloud interoperability (C2CI). In particular, security and mobility across organizational boundaries and domains are important attributes that need to be considered when assessing the maturity level of C2CI, especially from a usability and acceptability-for-use perspective. In [8], we extended the PAID attributes and presented a five-level maturity model for Cloud-to-Cloud interoperability.

6.1 Extension to PAID

The procedures (P) attribute will also reflect the availability of and adherence to uniform security and privacy policies and procedures that can be applied consistently across cloud boundaries, industrial standards for SLAs, standard procedures for cloud-services auditing, and technical and system architecture standards for cloud infrastructure and applications.

The application (A) attribute will also reflect the ease of cloud-service integration, as well as the ability of the software applications to work and migrate seamlessly across cloud boundaries while maintaining the same quality-of-service (QoS) levels.

The infrastructure (I) attribute will also reflect the degree of cloud mobility, availability of uniform tool sets for security (e.g., identity management), and cloud-services provisioning, management, monitoring, reporting and auditing.

The data (D) attribute will also reflect the degree of the evolution from an application-centric to a data-centric view of information processing. Instead of today's artificial separation between data and applications, information in the cloud will be treated as artifacts, which are embodiments of data and their associated manipulators, mini programs that allow the user to process (e.g., view, edit, and print) the data [9]. Manipulators are dynamically configured and associated with an artifact, according to the artifact's state, and can provide access and security control.

6.2 The Five-level C2CI Model

Since cloud computing builds on the premise that computing resources can be rapidly provisioned and released over the Internet, we can safely assume that, for any enterprise that is ready to migrate services to a cloud provider, the enterprise has surpassed levels 0 and 1 of the original LISI model and achieved the necessary networking and security maturity (e.g., protection of local area networks with firewalls and access control through local user authentication and file-access privileges) required to reach level 2. By removing Level 0, 1 and 2 from LISI, and adding three additional levels based on the degree of portability/mobility, security/privacy interoperability, ease of integration, and the availability of standard management, monitoring and audit

procedures and tools, we maintain the components of LISI that are applicable to the cloud model while adding the appropriate levels necessary for evaluating C2CI. The proposed C2CI model consists of the following levels:

(1) Level 0 – Domain-based interoperability in an integrated environment characterized by wide-area networks, shared data, separate applications, shared databases, and sophisticated collaboration. Cloud services are confined to single provider clouds.

(2) Level 1 – Enterprise-based interoperability in a universal environment characterized by wide-area networks, shared data, shared applications, cross-domain information sharing, and advanced collaborations via the inter-cloud Web services.

(3) Level 2 – Portability interoperability in a public, private, or hybrid cloud environment where cloud artifacts may traverse multiple providers in down states. Inter-cloud enforcement of security and privacy policies and SLA are based on pair wise agreements.

(4) Level 3 – Security interoperability in a public, private, or hybrid cloud environment where policies and procedures from one cloud provider will interact with other policies and procedures with other cloud provider(s) transparently and automatically using standardized protocols and cloud-wide formal trust relationships.

(5) Level 4 – Mobile interoperability in a public, private, or hybrid cloud environment where cloud artifacts may traverse multiple providers in in-flight states. There is no artificial separation between data and applications. Data in the cloud can be shared and manipulated by multiple applications on multiple platforms.

6.3 Applying the C2CI Model

At minimum, the tactical cloud needs to achieve C2CI Level 1 to support spectrum management. Tactical cloud computing can rely on solutions provided by emerging Web 3.0 standards, semantic web technologies for service registry and discovery, and SOA-based implementation to solve the data interoperability problems across organization boundaries. However, matured security interoperability at C2CI Level 3 is needed for the tactical cloud to guarantee the desired "quality of security service" discussed in Section 4.

7. CLOUD ADOPTION AND TRANSITION CONSIDERATIONS

To attain the full benefit of leveraging the cloud in the tactical environment, we must have a well laid-out plan for the transition from the current non-cloud environment to the future cloud-based tactical environment. The plan must spell out a smooth transitional path so that the tactical environment would continue to operate during the transition process. In [10], we presented a cloud service-adoption process that includes

research to identify potential cloud services, pathfinder experiments to overcome impediments, and pilot studies to create policy, an architecture, and a roadmap for their adoption (see Figure 1). Moreover, the process must provide the adoption teams with adequate decision points to drop or promote investigations of a particular cloud computing technology.

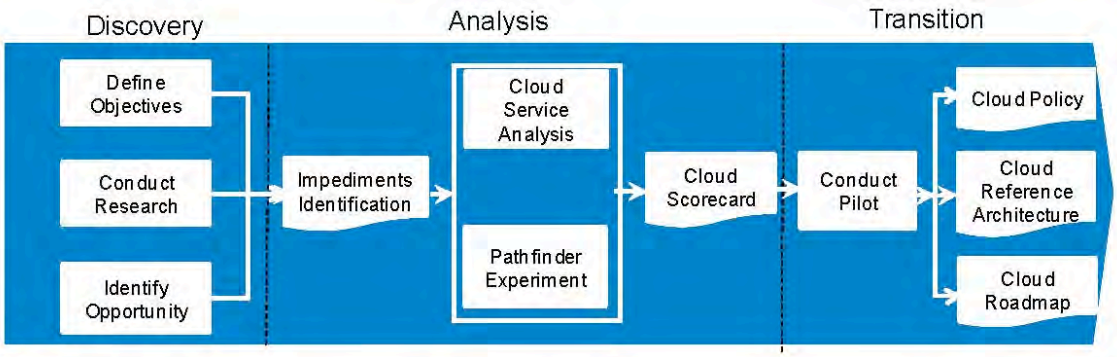

Figure 1. Cloud Service Adoption Process

8. CONTINUITY OF OPERATIONS CONSIDERATIONS

While the redundant nature of cloud computing is an enabler for robust, reliable tactical computing, the need for tactical systems to continue to operate even in the presence of one or more failures of cloud services necessitates new ways to architect the Spectrum-Management system, as demonstrated by the large-scale Amazon Web Services (AWS) blackout that occurred in April 2011 [11].

9. OPTIMIZATION ACROSS DIMENSIONS BEYOND SPECTRUM MANAGEMENT

Spectrum management should not be considered in isolation of engineering considerations, such as providing for elasticity by dynamically (i.e., at runtime) allocating the computing tasks between mobile devices and the cloud, taking into consideration factors such as the device's status, cloud's status, and user- and application-specified QoS, resulting in what Zhang *et al*. refer to as an "execution configuration" [12]. Zhang *et al*. also developed a cost model and algorithms for use in optimizing the execution environment, which takes into consideration based on real-time monitoring of

> Device and cloud related data such as battery level, network conditions, device loads, cloud loads and other performance data including current latency of the application….

They admitted however that for applications that are composed of "multiple types of weblets [components of an application that can be individually launched on the device or in the cloud], each having different runtime behaviors, the optimization can be very complex and the computation itself may override the cost savings" in offline or remote execution of the components.

In addition to the architectural framework introduced by Zhang et al., Rodriguez-Martinez *et al*. [13] and Damm, Rigz, and Strauch [14] have proposed software patterns and archetype patterns for use in developing elastic mobile cloud applications.

10. CONCLUSION

This report presents two high-level use cases to illustrate how cloud computing can play a role in supporting spectrum management—an important component of tactical communications. It also discusses the challenges and possible approaches for making the cloud-based system timely, robust, trustworthy, and interoperable, as well as means to validate and verify such systems.

10. REFERENCES

[1] A. K. Cebrowski and J. J. Garstka, "Network Centric Warfare: Its Origins and Future," *Proc. US Naval Institute*, 124(1), pp. 28-35, Jan. 1998.

[2] D. Alberts, J. Garstka, R. Hayes, and D. Signori, *Understanding Information Age Warfare*, CCRP Publications, September 2001. Accessed on 30 Oct. 2011: http://www.dodccrp.org/files/Alberts_UIAW.pdf

[3] D. Alberts and R. Hayes, *Power to the Edge*, CCRP Publications, Sept. 2003. Accessed on 30 Oct. 2011: http://www.dodccrp.org/files/Alberts_Power.pdf

[4] K. D. Foster, J. J. Shea, J. B. Michael, T. W. Otani, L. Peitso, and M. Shing, "Cloud Computing for Large-Scale Weapon Systems," in *Proc. 2010 IEEE International Conference on Granular Computing*, San Jose, Calif., 14-16 August 2010, pp. 161-166.

[5] J. Dixon, Integrating Cellular Handset Capabilities with Marine Corps Tactical Communications, Master's thesis, Naval Postgraduate School, Monterey, Calif., 2010. Accessed on 30 Oct. 2011: http://www.acquisitionresearch.net/_beta/files/FY2010/NPS-AM-10-004.pdf

[6] D. Drusinsky, J. B. Michael, and M. Shing, "A Visual Tradeoff Space for Formal Verification and Validation Techniques," *IEEE Systems Journal*, 2(4), pp. 513-519, Dec. 2008.

[7] C4ISR Architecture Working Group Interoperability Panel, Levels of Information Systems Interoperability (LISI), Department of Defense, Washington, D.C., 30 Mar. 1998.

[8] S. Dowell, A. Barreto III, J. B. Michael, and M. Shing, "Cloud to Cloud Interoperability," in *Proc. 6th International Conference on System of Systems Engineering*, Albuquerque, N.M., 27-30 June 2011, pp. 258-263.

[9] K. D., Foster, J. J. Shea, D. Drusinsky, J. B. Michael, T. W. Otani and M. Shing, "Removing the Boundaries: Steps toward a Cloud Nirvana," in *Proc. 2010 IEEE*

International Conference on Granular Computing, San Jose, Calif., 14-16 August 2010, pp. 167-171.

[10] S. Dowell, L. Peitso and M. Shing, "Cloud Service Adoption Process–An Approach for Developing Effective DoD Cloud Computing Strategies," accepted for publication in *Crosstalk* magazine.

[11] A. Cockroft, C. Hicks and G. Orzell, "Lessons Netflix Learned from the AWS Outage," The Netflix "Tech" Blog, 29 Apr 2011. Accessed on 30 Oct. 2011: http://techblog.netflix.com/2011/04/lessons-netflix-learned-from-aws-outage.html

[12] X. Zhang, A. Kunjithapatham, S. Jeon, and S. Gibbs, "Towards an Elastic Application Model for Augmenting the Computing Capabilities of Mobile Devices with Cloud Computing, *Mobile Networks and Applications* 16(3), pp. 270-284, 2011.

[13] M. Rodriguez-Martinez, J. Seguel, M. Sotomayor, J. P. Aleman, J. Rivera, and M. Greer, "Open911: Experiences with the Mobile Plus Cloud Paradigm," in *Proc. 4th IEEE International Conference on Cloud Computing*, Washington, D.C., 4-9 July 2011, pp. 606-613.

[14] S. Damm, T. Ritz, and J. Strauch, "Adaption of Archetype Patterns for Mobile Cloud-based Business Apps, in *Proc. 1st IEEE PerCom Workshop on Pervasive Communities and Service Clouds*, Seattle, Wash., 21-25 Mar. 2011, pp. 100-105.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Research Sponsored Programs Office, Code 41
   Naval Postgraduate School
   Monterey, CA 93943

4. Mr. John Snevely
   Office of the Under Secretary of Defense for Intelligence
   – Intelligence, Surveillance, and Reconnaissance
   5000 Defense Pentagon
   Washington, DC 20301-5000

5. Professor George Dinolt
   Naval Postgraduate School
   Monterey, California

6. Professor Doron Drusinsky
   Naval Postgraduate School
   Monterey, California

7. Professor Bret Michael
   Naval Postgraduate School
   Monterey, California

8. Professor Thomas Otani
   Naval Postgraduate School
   Monterey, California

9. Senior Lecturer Loren Peitso
   Naval Postgraduate School
   Monterey, California

10. Professor Man-Tak Shing
    Naval Postgraduate School
    Monterey, California

11. Mr. John Shea
    Office of the DoD CIO
    Alexandria, Virginia

12. CDR Kurt Rothenhaus
    PEO C4I/PMW 160
    San Diego, California

13. Mr. Tao Rocha
SPAWAR Atlantic
Charleston, South Carolina